

## 4.6. SISTEMAS DISTRIBUIDOS DE TIEMPO REAL

Los sistemas tolerantes de fallas no son el único tipo de sistemas distribuidos especializados. Los sistemas de tiempo real forman otra categoría. A veces se combinan estos dos tipos para obtener sistemas de tiempo real tolerantes de fallas.

### ¿Qué es un sistema de tiempo real?

Para la mayoría de los programas, el hecho de que sean correctos depende sólo de la secuencia lógica de las instrucciones ejecutadas, no del momento en que se ejecuten. Por el contrario, los programas (y sistemas) de tiempo real interactúan con el mundo exterior de una manera que implica al tiempo. Cuando aparece un estímulo, el sistema responde a éste de cierta manera y antes de cierto momento límite. Si entrega la respuesta correcta, pero después del límite, se considera que el sistema está fallando. El momento en que se produce la respuesta es tan importante como aquello que produce.

Consideremos un ejemplo sencillo. Un reproductor de discos compactos de audio consta de un CPU que toma los bits que llegan del disco y los procesa para generar música.

Supongamos que el CPU es lo bastante rápido como para hacer el trabajo. Ahora imaginemos que un competidor decide construir un reproductor más barato mediante un CPU con un tercio de la velocidad del otro. Si éste guarda todos los bits de llegada y los reproduce con un tercio de la velocidad esperada, las personas se sobrecogerán con el sonido y si sólo reproduce cada tercer nota, la audiencia tampoco quedará satisfecha.

Muchas aplicaciones y sistemas de tiempo real son muy estructurados, mucho más que los sistemas distribuidos de propósito general. De manera típica, un dispositivo externo (tal vez un reloj) genera un estímulo para la computadora, la que entonces debe realizar ciertas acciones antes de un momento límite. Al terminar el trabajo solicitado, el sistema queda inactivo hasta que llega el siguiente estímulo.

Con frecuencia, los estímulos son periódicos, de modo que un estímulo ocurre de manera regular cada AT segundos, como una computadora en un televisor o video casetera, que recibe un cuadro nuevo cada 1/60 segundos. A veces, los estímulos son periódicos, lo que significa que son recurrentes, pero no regulares, como en la llegada de un avión al espacio aéreo de un controlador de tráfico aéreo. Por último, algunos estímulos son esporádicos (inesperados), como el sobrecalentamiento de un dispositivo. Aún en un sistema que en gran medida sea periódico, una complicación es que pueden existir muchos tipos de eventos, como entrada de video, entrada de audio y el control de la unidad motora, cada uno con su período y acciones necesarias. La figura 4-25 muestra una situación con tres flujos de eventos periódicos, A, B y C, más un evento esporádico, X

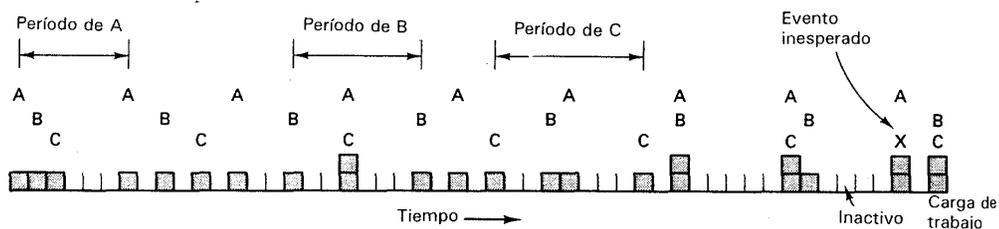


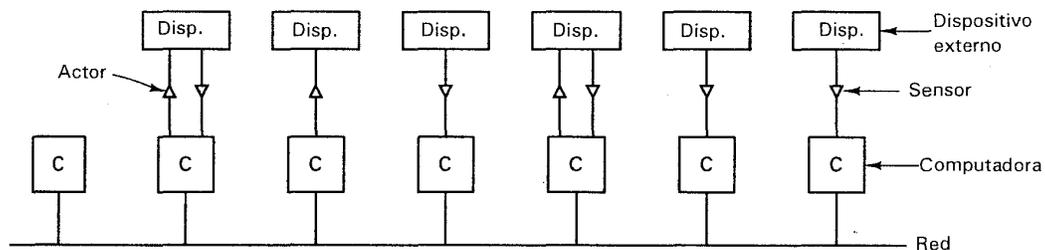
Figura 4-25. Superposición de tres flujos de eventos más un evento esporádico.

A pesar del hecho de que el CPU tiene que trabajar con varios flujos de eventos, no es aceptable que diga: Es cierto que omití el evento B, pero no es mi error; yo seguía trabajando en A cuando ocurrió B. Aunque no es difícil controlar dos o tres flujos de

entrada con interrupciones de prioridad, conforme las aplicaciones son cada vez mayores y complejas (por ejemplo, las líneas de ensamblaje en una fábrica automatizada con miles de robots), será cada vez más difícil que una máquina cumpla con todas las horas límite y otras restricciones de tiempo real.

En consecuencia algunos diseñadores están experimentando con la idea de colocar un microprocesador exclusivo al frente de cada dispositivo de tiempo real para aceptar salida de ella cuando tenga algo que decir, y dar una entrada con la velocidad que requiera. Por supuesto, esto no hace que el carácter de tiempo real se esfume, sino que da lugar a un sistema distribuido de tiempo real, con sus propias características y retos (por ejemplo, la comunicación de tiempo real).

Los sistemas distribuidos de tiempo real pueden estructurarse con frecuencia como se ilustra en la figura 4-26. Aquí vemos una colección de computadoras conectadas mediante una red. Algunas de éstas se conectan a dispositivos externos que producen o aceptan datos o esperar ser controlados en tiempo real. Las computadoras pueden ser pequeños micro controladores integrados a los dispositivos, o máquinas independientes. En ambos casos, por lo general tienen sensores para recibir señales de los dispositivos y/o actores a los cuales enviar señales. Los sensores y actores pueden ser digitales o analógicos.



**Figura 4-26.** Un sistema de cómputo distribuido de tiempo real.

Los sistemas de tiempo real se clasifican por lo general en dos tipos dependiendo de lo serio de sus tiempos límite y de las consecuencias de omitir uno de ellos. Estos son:

1. Sistemas de tiempo real suave.
2. Sistemas de tiempo real duro.

El tiempo real suave significa que no existe problema si se rebasa un tiempo límite. Por ejemplo, un conmutador telefónico bajo condiciones de sobrecarga podría perder o equivocar de ruta IO llamadas y seguir cumpliendo sus especificaciones. Por el contrario, un tiempo límite no cumplido en un sistema de tiempo real duro es inaceptable, pues podría conducir a la pérdida de una vida o a una catástrofe ambiental. En la práctica, existen también sistemas intermedios en los que la omisión de un tiempo límite significa que falla toda la actividad actual, pero que la consecuencia no es fatal. Por ejemplo, si una botella de refresco ha pasado por la banda de conducción sin detenerse en la boquilla de alimentación, no hay necesidad de continuar vertiendo refresco en ella, pero los resultados no son fatales. Además, e algunos sistemas de tiempo real, algunos subsistemas son de tiempo real duro, mientras que otros son de tiempo real suave.

Los sistemas de tiempo real han estado por ahí durante décadas, de modo que existe mucha sabiduría popular acumulada sobre ellos, pero muchos de ellos son incorrectos.

Stankovic (1988) ha señalado algunos de éstos, y aquí resumimos los peores.

Mito 1: Los sistemas de tiempo real tratan de la escritura de controladores de dispositivos en código ensamblador.

Tal vez esto era cierto en la década de 1970 para los sistemas de tiempo real que constaban de unos cuantos instrumentos conectados a una mini computadora, pero los sistemas de tiempo real de la actualidad son muy complicados para confiar en el lenguaje ensamblador y escribir los controladores de dispositivos es la menor de las preocupaciones de un diseñador de sistemas de tiempo real, Mito 2.' El cómputo de tiempo real es rápido. No necesariamente. Un telescopio controlado por una computadora puede tener que rastrear las estrellas o galaxias en tiempo real, pero la aparente rotación del cielo es de tan sólo 15 grados de arco por hora de tiempo, que no es en particular rápido. Aquí lo que cuenta es la exactitud.

Mito 3: Las computadoras rápidas harán que el sistema de tiempo real sea obsoleto.

No. Sólo animan a las personas a construir sistemas de tiempo real que anteriormente estaban más allá de lo normal. Los cardiólogos estarían felices de tener un digitalizador MR! que muestre el latido de un corazón dentro de un paciente que realiza ejercicio en tiempo real. Cuando lo tengan, pedirán uno en tres dimensiones, a color, y con la posibilidad de realizar acercamientos o alejamientos. Además, para crear sistemas más rápidos mediante el uso de varios procesadores hay que resolver nuevos problemas de comunicación, sincronización y planificación.

#### **Aspectos del diseño**

Los sistemas distribuidos de tiempo real tienen ciertos aspectos de diseño únicos.

#### **Sincronización del reloj**

El primer aspecto es el mantenimiento de la propia hora. Con varias computadoras, cada una con su propio reloj local, la sincronización de los relojes es un aspecto fundamental.

#### **Sistemas activados por eventos vs. sistemas activados por el tiempo**

En un sistema de tiempo real activado por eventos, cuando ocurre un evento significativo en el mundo exterior, es detectado por algún sensor, lo que entonces provoca que el CPU conectado tenga una interrupción. Los sistemas activados por eventos están controlados entonces por las interrupciones. La mayor parte de los sistemas de tiempo real funcionan de esta manera. Para los sistemas de tiempo real suave con mucho poder de cómputo por compartir, este método es sencillo, funciona bien y sigue utilizándose con amplitud. Aún para sistemas más complejos, funciona bien si el compilador puede analizar el programa y conoce todo lo que hay que conocer acerca del comportamiento del sistema una vez que ocurra un evento, aunque no pueda decir el momento en que ocurrirá el evento. El principal problema con los sistemas activados por eventos es que pueden fallar bajo condiciones de carga pesada, es decir, cuando muchos eventos ocurran a la vez.